Paul had a birthday yesterday, and they were playing a guessing game there with Andrew: Andrew was trying to guess Paul's age. Andrew knew that Paul's age is an integer between 1 and n, inclusive. Andrew can guess any number x between 1 and n, and Paul will tell him what is the greatest common divisor of x and his age.

Here's a possible course of the game for $n = 6$. Andrew starts with guessing 3, and Paul replies that the greatest common divisor of 3 and his age is 1. That means that Paul's age can't be 3 or 6, but can still be 1, 2, 4 or 5. Andrew continues with guessing 2, and Paul replies 2. That means that Paul's age can't be 1 or 5, and the only two remaining choices are 2 and 4. Finally, Andrew guesses 4, and Paul replies 2. That means that Paul's age is 2, and the game is over.

Andrew needed three guesses in the above example, but it's possible to always determine Paul's age in at most two guesses for $n = 6$. The optimal strategy for Andrew is: at the first step, guess 6. If Paul says 1, then its 1 or 5 and he can check which one by guessing 5. If Paul says 2, then its 2 or 4, and he can check by guesing 4 as we've seen above. If Paul says 3, then we already know the answer is 3. Finally, if Paul says 6, the answer is 6.

What is the number of guesses required in the worst case if Andrew guesses optimally for the given n?

Input

The input will contain several test cases, each of them as described below.

The input file contains one integer n, $2 \leq n \leq 10000$.

Output

For each test case, write to the output on a line by itself.

Output one integer — the number of guesses Andrew will need to make in the worst case.

Sample Input

```
6
```

Sample Output

```
2
```